



 Latest updates: <https://dl.acm.org/doi/10.1145/3746027.3755754>

RESEARCH-ARTICLE

## **Input Domain Aware MoE: Decoupling Routing Decisions from Task Optimization in Mixture of Experts**

**YONGXIANG HUA**, University of Science and Technology of China, Hefei, Anhui, China

**HAOYU CAO**, University of Science and Technology of China, Hefei, Anhui, China

**ZHOU TAO**, University of Science and Technology of China, Hefei, Anhui, China

**BOCHENG LI**, University of Science and Technology of China, Hefei, Anhui, China

**ZIHAO WU**, University of Science and Technology of China, Hefei, Anhui, China

**CHAOHU LIU**, University of Science and Technology of China, Hefei, Anhui, China

[View all](#)

**Open Access Support** provided by:

**University of Science and Technology of China**



PDF Download  
3746027.3755754.pdf  
23 December 2025  
Total Citations: 0  
Total Downloads: 53

Published: 27 October 2025

[Citation in BibTeX format](#)

MM '25: The 33rd ACM International Conference on Multimedia  
October 27 - 31, 2025  
Dublin, Ireland

Conference Sponsors:  
SIGMM

# Input Domain Aware MoE: Decoupling Routing Decisions from Task Optimization in Mixture of Experts

YongXiang Hua

University of Science and Technology  
of China  
State Key Laboratory of Cognitive  
Intelligence  
Hefei, China  
yx15333063290@mail.ustc.edu.cn

Haoyu Cao

University of Science and Technology  
of China  
State Key Laboratory of Cognitive  
Intelligence  
Hefei, China  
caohaoyu@mail.ustc.edu.cn

Zhou Tao

University of Science and Technology  
of China  
State Key Laboratory of Cognitive  
Intelligence  
Hefei, China  
zhoutao24@mail.ustc.edu.cn

Bocheng Li

University of Science and Technology  
of China  
State Key Laboratory of Cognitive  
Intelligence  
Hefei, China  
bcli@mail.ustc.edu.cn

Zihao Wu

University of Science and Technology  
of China  
State Key Laboratory of Cognitive  
Intelligence  
Hefei, China  
zihawu@mail.ustc.edu.cn

Chaohu Liu

University of Science and Technology  
of China  
State Key Laboratory of Cognitive  
Intelligence  
Hefei, China  
liuchahu@mail.ustc.edu.cn

Linli Xu\*

University of Science and Technology  
of China  
State Key Laboratory of Cognitive  
Intelligence  
Hefei, China  
linlixu@ustc.edu.cn

## Abstract

Sparse Mixture of Experts (sMoE) has become a pivotal approach for scaling large vision-language models, offering substantial capacity while maintaining computational efficiency through dynamic, sparse activation of experts. However, existing routing mechanisms, typically based on similarity scoring, struggle to effectively capture the underlying input structure. This limitation leads to a trade-off between expert specialization and balanced computation, hindering both scalability and performance. We propose Input Domain Aware MoE, a novel routing framework that leverages a probabilistic mixture model to better partition the input space. By modeling routing probabilities as a mixture of distributions, our method enables experts to develop clear specialization boundaries while achieving balanced utilization. Unlike conventional approaches, our routing mechanism is trained independently of task-specific objectives, allowing for stable optimization and decisive expert

assignments. Empirical results on vision-language tasks demonstrate that our method consistently outperforms existing sMoE approaches, achieving higher task performance and improved expert utilization balance.

## CCS Concepts

- Computing methodologies → Computer vision; Natural language processing; Ensemble methods.

## Keywords

Large Multimodal Model; Visual Question Answering; Mixture of Experts; Load Balancing

## ACM Reference Format:

YongXiang Hua, Haoyu Cao, Zhou Tao, Bocheng Li, Zihao Wu, Chaohu Liu, and Linli Xu. 2025. Input Domain Aware MoE: Decoupling Routing Decisions from Task Optimization in Mixture of Experts. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25)*, October 27–31, 2025, Dublin, Ireland. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3746027.3755754>

\*Corresponding author

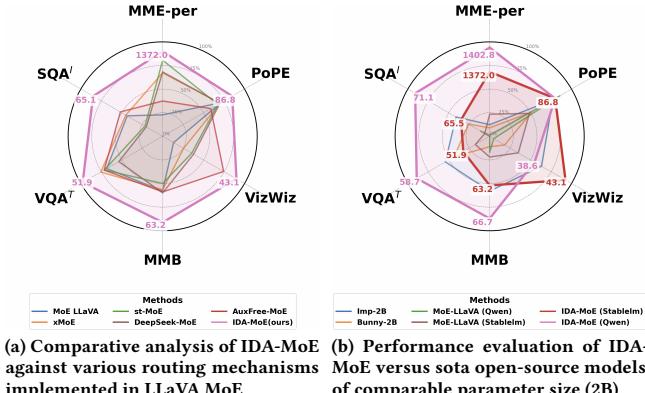
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '25, Dublin, Ireland

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 979-8-4007-2035-2/2025/10  
<https://doi.org/10.1145/3746027.3755754>

## 1 Introduction

Vision-language models (VLMs) have demonstrated remarkable capabilities, but their immense computational demands pose significant scaling challenges. To address this, sparsely-gated Mixture-of-Experts (sMoE) was introduced in Large Language Models (LLMs) [22] and has since established itself as a dominant solution, finding widespread applications across models of varying scales and



**Figure 1: Benchmark performance of IDA-MoE (ours) across visual question answering and multimodal reasoning tasks. The left panel (Fig. 1a) illustrates the effectiveness of different MoE routing implementations, while the right panel (Fig. 1b) demonstrates comparative performance against models of similar computational scale.**

domains[4, 8, 13, 34, 58, 63, 65]. At its core, sMoE employs conditional computation where each input token activates only a subset of specialized experts through a learned routing function, effectively reducing computational overhead while maintaining model capacity. However, the effectiveness of this approach ultimately hinges on the routing mechanism that assigns inputs to experts.

Various routing methods have been proposed. Among them, the predominant approaches [8, 10, 22, 32, 39] rely on similarity-based scoring, where routing probabilities are computed through softmax over token-expert alignment scores. This mechanism, trained end-to-end with the task objective, tends to develop a strong bias towards routing tokens to high-performing experts, creating a feedback loop that exacerbates load imbalance. To mitigate this, conventional approaches introduce auxiliary load balancing objectives[22, 47, 70] that push per-token routing distributions toward uniformity across experts. However, this creates a fundamental contradiction: while the core principle of sMoE emphasizes specialized experts through conditional computation, load balancing forces indiscriminate token distribution, inevitably undermining expert specialization. This unresolved tension manifests in two critical problems: during training, routing instability leads to frequent changes in token-expert assignments, severely impacting sample efficiency [9, 70]; during inference, ambiguous routing decisions result in suboptimal expert selection, compromising model performance and robustness to distribution shifts [53, 57].

The root cause lies in coupling routing decisions with task optimization, which makes it contradictory to achieve balanced token allocation while preserving sharp, decisive routing. Specifically, when experts are evaluated solely on their contribution to the final loss, the system naturally concentrates computation on a small subset of versatile experts rather than developing complementary specializations(analysed in Section 3.2). This suggests a fundamental need to decouple routing from task-specific learning signals and instead ground it in the inherent structure of the input distribution.

To this end, we propose Input Domain Aware MoE (IDA-MoE), which explicitly models the input space through a Gaussian Mixture Model (GMM) to create natural, data-driven routing boundaries. This probabilistic framework partitions inputs based on their distribution characteristics rather than loss contributions, where routing decisions are guided by the posterior probabilities rather than learned similarity scores, enabling experts to develop clear specializations while maintaining balanced utilization. By associating each expert with multiple mixture components (i.e., regions of the input distribution) rather than learning routing directly from task objectives, our approach creates a flexible mapping between input domains and expert specializations. This decoupling allows independent optimization of expert workload distribution (controlled by mixture priors  $P(k)$ ) while maintaining sharp, distribution-aware token routing - achieving load balance as an emergent property rather than an enforced constraint. Our contributions in this work are threefold.

- First, we present a systematic analysis of existing MoE routing mechanisms, revealing the fundamental dilemma: the coupling between routing and task objectives inherently drives load imbalance, while conventional mitigation strategies often compromise training stability and prevent effective expert specialization.
- Second, we introduce Input Domain Aware MoE (IDA-MoE), a novel architecture that explicitly models input distribution through probabilistic mixture modeling, effectively decoupling routing decisions from task optimization while maintaining precise expert specialization.
- Third, through comprehensive experiments across multiple vision-language tasks, we demonstrate that IDA-MoE achieves superior performance compared to traditional routing approaches, exhibiting improved training stability, better expert utilization, and enhanced performance on VQA benchmarks.

## 2 Related Works

*Large Vision-Language Models.* The emergence of powerful Large Language Models (LLMs) has enabled a new paradigm for multimodal learning, where frozen LLMs act as reasoning engines for vision-language tasks. Early approaches like BLIP-2[24] and LLaVA[37, 38] proposed lightweight adapters (e.g., Q-Former, MLP projectors) to map visual features from pretrained encoders (e.g., CLIP[43]) into the LLM’s token space. This adapter-based paradigm has achieved substantial performance gains across diverse vision-language tasks. Recent works [2, 5, 6, 11, 12, 14, 48, 62] continue this paradigm by aggressively scaling both model parameters and training data. While this leads to improved capabilities, it also makes training and inference prohibitively expensive. To address these computational challenges, researchers have explored sparse Mixture of Experts (sMoE) architectures, which are adopted both in pre-training [2, 3, 59] and via efficient upcycling of dense models [21, 25, 27, 30, 50, 56, 60] in multimodal context. Despite these advances, sMoE architectures face significant load balancing problems. This challenge is particularly pronounced in multimodal contexts, as inputs from the same modality tend to activate identical expert pathways, further exacerbating the load imbalance problem.

*Mixture-of-Experts*, particularly Sparse Mixture-of-Experts, was introduced to large language models by Gshard[22] and refined in subsequent works [8, 10, 19, 32, 55, 61]. This architecture enables capacity scaling while preserving computational efficiency through selective expert activation. Since its introduction, sparse Mixture-of-Experts models have faced persistent challenges with the load imbalance problem. The predominant approaches rely on **Auxiliary Losses**, which penalize imbalanced expert utilization based on token allocation fractions[10, 22, 32], Coefficient of Variance of token allocation[47] or routing entropy[49]. However, multiple studies [9, 53, 57] demonstrate that auxiliary losses can significantly impair training stability and model performance. st-MoE[70] introduced Router-z Loss to stabilize training. Recent approaches such as Auxiliary-free-MoE[53] and Deepseek-v3[33] attempt to mitigate the side effects of load balancing loss through introducing prior distribution correction. Another line of work achieves **Runtime Load Balancing** via token dropping [10] or sophisticated system-level approaches that involve dynamic resource allocation or predictive scheduling[16, 20, 42, 54]. Nevertheless, these methods have yet to achieve an optimal balance between computational efficiency and model performance in multimodal settings. In contrast to previous approaches, our work addresses the fundamental tension between expert specialization and computational balance by introducing a novel routing mechanism that enables effective specialization while maintaining efficient resource utilization.

### 3 The Specialization-Balance Dilemma

In this section, we delve into a fundamental challenge of standard Sparse Mixture-of-Experts(sMoE) architectures: the inherent tension between achieving expert specialization and maintaining balanced computation. We first provide the necessary background on the standard sMoE architecture and token-choice gating mechanism (Section 3.1). Then, we analyze how coupling routing decisions directly with task optimization promotes expert specialization but results in inevitable load imbalance (Section 3.2). Finally, we examine how conventional auxiliary load balancing loss compromises specialization and stability, thus revealing the core dilemma (Section 3.3).

#### 3.1 Sparse Mixture of Experts

In modern transformer architectures, Sparse Mixture of Experts (sMoE) represents a powerful approach to scaling model capacity while maintaining computational efficiency. sMoE typically replaces the standard feedforward network (FFN) with a set of expert networks, where each input token is dynamically routed to a small subset of these experts based on its content.

Consider a standard  $L$ -layer Transformer processing an input sequence  $X \in \mathbb{R}^{T \times d}$  with  $T$  tokens and embedding dimension  $d$ . At each layer  $l$ , let  $\mathbf{h}_t^l$  represent the hidden state for token  $t$ , and  $\mathbf{u}_t^l$  represent the intermediate representation after self-attention:

$$\mathbf{u}_t^l = \text{SelfAttn}(\mathbf{h}_t^{l-1}) + \mathbf{h}_t^{l-1} \quad (1)$$

$$\mathbf{h}_t^l = \text{FFN}(\mathbf{u}_t^l) + \mathbf{u}_t^l \quad (2)$$

In sMoE-based transformers, the FFN module is replaced with a dynamic mixture of expert networks:

$$\mathbf{h}_t^l = \sum_{i=1}^k \mathcal{G}(\mathbf{u}_t^l, \mathbf{e}_i) \cdot \text{FFN}_i(\mathbf{u}_t^l) + \mathbf{u}_t^l \quad (3)$$

where  $\mathcal{G}(\cdot, \mathbf{e}_i)$  computes routing scores measuring token-expert compatibility, with  $\mathbf{e}_i$  representing routing parameters of expert  $i$ , and  $\text{FFN}_i(\cdot)$  represents the  $i$ -th expert network. For simplicity, we note  $\text{MoE}(\mathbf{u}_t^l) = \sum_{i=1}^k \mathcal{G}(\mathbf{u}_t^l, \mathbf{e}_i) \cdot \text{FFN}_i(\mathbf{u}_t^l)$ .

The choice of gating function is crucial for dynamic expert selection. The predominant approach in large language and vision-language models is Token-Choice Gating, where each token independently selects its preferred top- $k$  experts based on compatibility scores. While alternative routing strategies exist, they present various trade-offs. For instance, Expert-Choice Gating [67, 68] enforces uniform expert workload but risks uneven token coverage and violates the causal constraints of language modeling [53], while Non-trainable Gating [44, 45, 71] offers balanced computation via static or randomized routing at the cost of adaptability. Considering these trade-offs, our work focuses on Token-Choice Gating, which computes routing weights as:

$$\mathbf{p}_t = \text{Softmax}(\mathbf{u}_t^{l-1} \cdot [\mathbf{e}_1, \dots, \mathbf{e}_K]) \quad (4)$$

$$\mathcal{G}(\mathbf{u}_t^{l-1}, \mathbf{e}_i) = \delta_{t,i} \cdot \mathbf{p}_{t,i} \quad (5)$$

where  $\mathbf{p}_t$  represents the routing distribution for token  $t$ , with each element  $\mathbf{p}_{t,i}$  measuring the token-expert compatibility.  $\delta_{t,i}$  is a binary indicator, which equals 1 only when expert  $i$  is among the top- $k$  highest-probability experts for token  $t$ .

#### 3.2 Load Imbalance

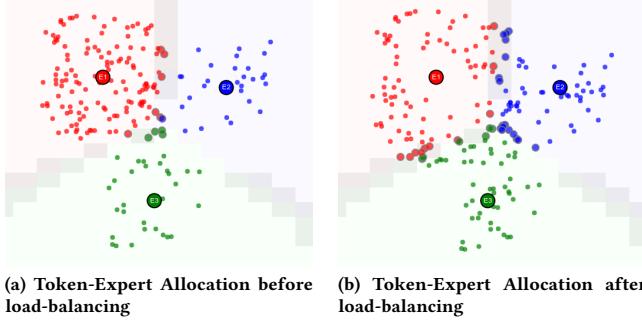
Load imbalance is a persistent and critical challenge in sMoE models. Numerous studies [8, 10, 22, 47, 53, 68] observe that computation is often heavily skewed in practice, with a small subset of experts processing a disproportionate fraction of tokens while others remain largely inactive. This disparity significantly compromises parameter efficiency, computation utilization, and the overall model scaling potential. While previous works have attempted to mitigate this issue through auxiliary load balancing losses (discussed in Section 3.3), we demonstrate that load imbalance is not merely an implementation side-effect, but an inherent consequence of the underlying sMoE training dynamics. By analyzing how the task loss  $\mathcal{L}$  influences the expert parameters  $\mathbf{e}_i$ , which in turn determine routing probabilities, we can reveal the fundamental mechanisms that drive this imbalance. Specifically, the gradient of the task loss with respect to  $\mathbf{e}_i$  is given by:

$$\nabla_{\mathbf{e}_i} \mathcal{L} = \sum_{t=1}^T \delta_{t,i} \mathbf{p}_{t,i} \cdot (\mu_{t,i} - \bar{\mu}_t) \cdot \mathbf{u}_t^{l-1} \quad (6)$$

$$\mu_{t,i} = \frac{\partial \mathcal{L}}{\partial \text{MoE}(\mathbf{u}_t^{l-1})} \cdot \text{FFN}_i(\mathbf{u}_t^{l-1}) \quad (7)$$

$$\bar{\mu}_t = \frac{\partial \mathcal{L}}{\partial \text{MoE}(\mathbf{u}_t^{l-1})} \cdot \text{MoE}(\mathbf{u}_t^{l-1}) \quad (8)$$

where  $\mu_{t,i}$  quantifies how effectively expert  $i$  processes token  $t$  through task-specific loss gradients, while  $\bar{\mu}_t$  represents the average effectiveness of the activated experts for that token. This gradient



**Figure 2: Illustration of load balancing loss effects.** Experts colored circles and their corresponding token assignments colored dots in matching colors. The transition from Fig. 2a to Fig. 2b simulates the effect of load balancing loss. Token embeddings are pushed toward and accumulated around decision boundaries.

dictates updates to the routing parameters  $\mathbf{e}_i$ , thereby shaping future routing probabilities  $\mathbf{p}_{t',i}$  for similar tokens  $t'$ .

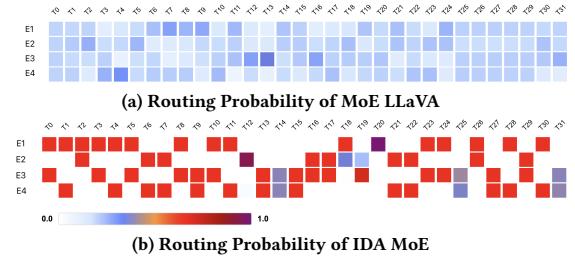
Careful examination of this gradient reveals several interconnected factors that systematically drive expert imbalance. 1) **Relative Performance Steering**: Updates are steered by the relative performance metric  $(\mu_{t,i} - \bar{\mu}_t)$ , and  $\mathbf{e}_i$  is preferentially updated towards alignment with tokens where expert  $i$  outperforms its peers in loss reduction. This leads to expert specialization but also concentrates tokens around already-competent experts. 2) **Probability Amplification**: The expert concentration is amplified by the probability amplification factor  $\mathbf{p}_{t,i}$  in Eq. (6), as experts already deemed more suitable for certain tokens receive proportionally stronger gradient updates, creating a multiplicative effect that rapidly amplifies small initial advantages. 3) **Hard Gating Exclusion**: The Top-K selection mechanism exacerbates the imbalance by completely blocking updates for non-selected experts. This prevents adaptation for less-utilized experts and can lead to permanently underutilized or “dead” experts, especially during distribution shifts in training.

In conjunction, the mechanisms of performance biased steering, probability amplified updates, and hard gating interact to create a powerful positive feedback loop that systematically reinforces initial imbalances across the expert pool. Experts gaining an early lead attract more tokens and stronger updates, solidifying their dominance, while less favored experts become progressively starved of both data and meaningful gradients, leading to severe and potentially irreversible load imbalance. This “rich-get-richer” dynamic demonstrates that load imbalance is fundamentally intertwined with optimizing routing directly via task performance feedback.

### 3.3 The Specialization-Balance Dilemma

To counteract the inherent load imbalance described previously, the standard practice in sMoE models involves incorporating an auxiliary load balancing loss [10, 19, 22, 30, 32]. One of the most prominent form is:

$$\mathcal{L}_{\text{Balance}} = \alpha \sum_{i=1}^N f_i \mathbf{p}_i \quad (9)$$



**Figure 3: Routing probability heatmaps of the last MoE layer for the same input sequence.** The x-axis represents token indices, the y-axis represents expert indices, and the color intensity indicates the probability of assigning the  $i$ -th token to the  $j$ -th expert. Note how IDA-MoE (b) exhibits significantly more decisive routing patterns with clearer token-expert affinities compared to the conventional approach (a).

$$f_i = \frac{N}{KT} \sum_{t=1}^T \delta_{t,i} \quad (10)$$

$$\mathbf{p}_i = \frac{1}{T} \sum_{t=1}^T \mathbf{p}_{t,i} \quad (11)$$

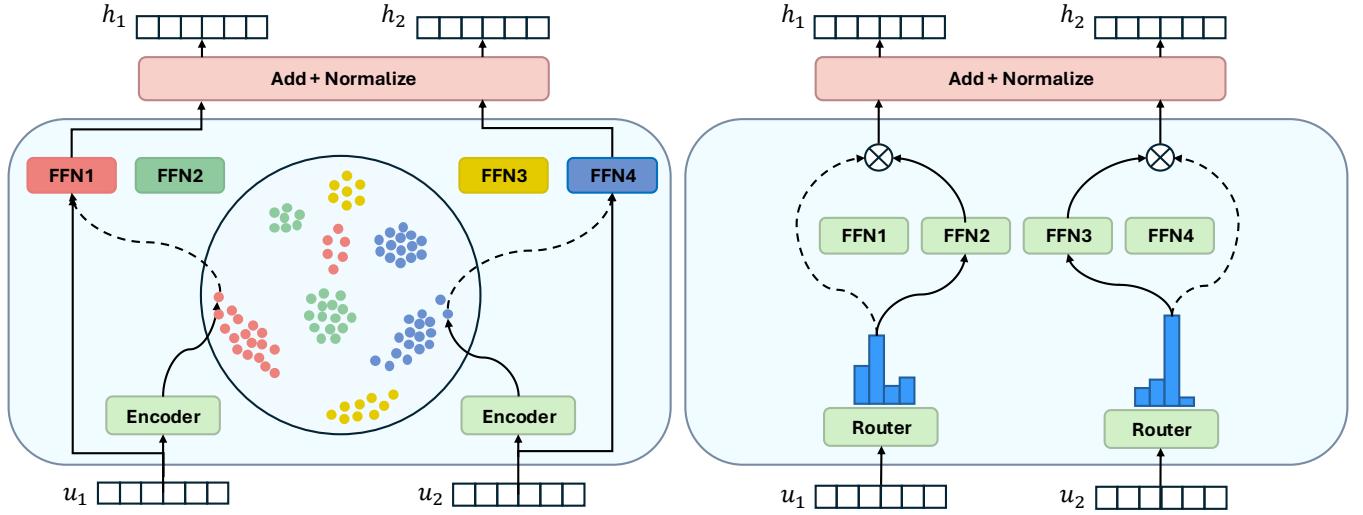
where  $N$  is the total number of experts,  $K$  is the number of experts selected for each token,  $f_i$  represents the fraction of tokens routed to expert  $i$ , and  $\alpha$  is a hyper-parameter controlling the strength of the auxiliary loss.

This balancing loss introduces a fundamental **specialization-balance dilemma**: while effective expert specialization requires the router to make decisive assignments through sharply peaked routing probabilities  $\mathbf{p}_{t,i}$  (directing each token confidently to its most appropriate experts), the auxiliary loss actively counteracts this tendency by penalizing concentrated allocations to achieve more balanced token distribution across the expert pool.

Specifically, auxiliary loss seeks to balance the workload  $f_i$  across experts primarily by penalizing high routing probabilities  $\mathbf{p}_{t,i}$ . As illustrated in Fig. 2, this loss redistributes tokens from overloaded experts to underutilized ones by pushing them across decision boundaries. However, this redistribution creates an unintended side effect: tokens tend to accumulate near routing decision boundaries (Fig. 2b), leading to ambiguous expert assignments and suboptimal performance during both training and inference.

Specifically, during training, tokens near boundaries may frequently switch assigned experts due to the competing pressures of task optimization and the load balancing loss, hindering both training stability and expert specialization. During inference, as illustrated in Fig. 3, auxiliary losses typically produce high-entropy routing distributions, where token-expert assignment probabilities appear less distinct compared to an ideal sharp routing (Fig. 3b). Such ambiguous routing creates inherent robustness issues. As prior research [53, 57] suggest, models with less decisive routing are more susceptible to performance degradation under input noise or distribution shifts, as small perturbations can easily alter routing decisions and consequently change model behavior.

Therefore, while conventional auxiliary losses successfully enforce load balance, they do so by fundamentally compromising the



**Figure 4: Comparison between IDA-MoE and traditional MoE routing.** (Left) IDA-MoE: Input token representations ( $u_1, u_2$ ) are projected into a lower-dimensional routing space where their distribution is modeled into distinct clusters. Tokens are routed to specific experts based on their cluster index, effectively decoupling routing decisions from task-specific optimization. (Right) Traditional MoE: A learned router module directly calculates token-expert affinities (classification-style) to determine expert assignments, typically trained end-to-end with the task objective.

model’s ability to specialize effectively. The resulting trade-off negatively impacts both training stability and inference robustness. This highlights the critical need for alternative routing strategies that can achieve balanced computation by directly addressing the input space structure, thereby preserving the sharpness and integrity of specialized expert routing.

## 4 Method

We propose Input Domain-Aware Mixture of Experts (IDA-MoE), a novel approach designed to achieve effective expert specialization and natural load balancing by decoupling routing decisions from task optimization and instead grounding them in the inherent structure of the input token distribution. Our method consists of three key components: (1) A Gaussian Mixture Model that captures the underlying distribution of input tokens in a low-dimensional space, (2) A component-based expert routing mechanism that leverages this probabilistic distribution model to make more informed and stable routing decisions, and (3) A component reactivation strategy that accelerates model convergence by addressing the uneven learning dynamics.

### 4.1 Decoupled Input Distribution Modeling

To efficiently model the input distribution while avoiding the curse of dimensionality, we first project the token representations  $u^{l-1}$  into a lower-dimensional latent space using a simple autoencoder network:

$$\mathbf{z}_t = \text{Encoder}(\text{sg}(\mathbf{u}^{l-1})) \quad (12)$$

where  $\text{sg}()$  stand for stop gradient. This encoder is trained concurrently with the main model using a reconstruction loss to preserve the underlying input structure:

$$\mathcal{L}_{\text{AE}} = \|\text{sg}(\mathbf{u}^{l-1}) - \text{Decoder}(\mathbf{z}_t)\|^2 \quad (13)$$

Within this compressed routing space, we model the input data distribution using a Gaussian Mixture Model (GMM). To enable experts to capture the intricacy of the underlying input structure, we associate the input domain of each expert with  $M$  fine-grained GMM components rather than a single Gaussian distribution. Thus, the overall distribution of the routing space can be formulated as:

$$p(\mathbf{z}_t) = \sum_{i=1}^N \sum_{m=1}^M \pi_{i,m} \mathcal{N}(\mathbf{z}_t | \mu_{i,m}, \Sigma_{i,m}) \quad (14)$$

where  $\pi_{i,m}$  denotes the mixing coefficient, and  $\mu_{i,m}, \Sigma_{i,m}$  are the mean and covariance of the  $m$ -th component for the  $i$ -th expert.

The GMM is trained jointly with the backbone model by minimizing the negative log-likelihood of the observed token representations under the mixture distribution:

$$\mathcal{L}_{\text{GMM}} = - \sum_{t=1}^T \log \left( \sum_{i=1}^N \sum_{m=1}^M \pi_{i,m} \mathcal{N}(\text{sg}(\mathbf{z}_t) | \mu_{i,m}, \Sigma_{i,m}) \right) \quad (15)$$

### 4.2 Component-Based Expert Routing

Standard MoE layers typically route each token to  $k \geq 1$  experts. To implement top- $k$  routing, IDA-MoE employs  $k$  independent sets of expert-associated GMM parameters as described in Section 4.1. Each GMM set  $j \in \{1, \dots, k\}$  contains parameter triplets  $(\pi_{j,i,m}, \mu_{j,i,m}, \Sigma_{j,i,m})$  and is trained using an independent NLL loss  $\mathcal{L}_{\text{GMM}_j}$ . This allows IDA-MoE to make  $k$  distinct expert selections for each token based on potentially different distributional perspectives captured by each GMM set. This implementation introduces minimal overhead due to the low dimensionality of  $\mathbf{z}_t$ . The routing process for a token  $\mathbf{u}_t$  with latent representation  $\mathbf{z}_t$  proceeds as follows:

**Calculate Posteriors:** For each selection rank  $j$ , we use the corresponding  $j$ -th GMM parameter set to compute the posterior probability  $P_j(i, m | \mathbf{z}_t)$ . This posterior quantifies the likelihood that the latent token  $\mathbf{z}_t$  belongs to component  $m$  associated with expert  $i$ :

$$P_j(i, m | \mathbf{z}_t) = \frac{\pi_{j,i,m} \mathcal{N}(\mathbf{z}_t | \boldsymbol{\mu}_{j,i,m}, \boldsymbol{\Sigma}_{j,i,m})}{\sum_{i'=1}^N \sum_{m'=1}^M \pi_{j,i',m'} \mathcal{N}(\mathbf{z}_t | \boldsymbol{\mu}_{j,i',m'}, \boldsymbol{\Sigma}_{j,i',m'})} \quad (16)$$

**Select Top Experts:** For each rank  $j$ , we determine the expert  $i_{t,j}^*$  whose components maximize the posterior probability for  $\mathbf{z}_t$ .

$$P_j^*(i, \mathbf{z}_t) = \max_{m \in \{1, \dots, M\}} P_j(i, m | \mathbf{z}_t) \quad \forall i \in \{1, \dots, N\} \quad (17)$$

$$i_{t,j}^* = \arg \max_{i \in \{1, \dots, N\}} P_j^*(i, \mathbf{z}_t) \quad (18)$$

This yields a set of  $k$  selected expert indices  $\mathcal{I}_t = \{i_{t,1}^*, \dots, i_{t,k}^*\}$  and a corresponding vector of their maximum posterior scores  $\mathbf{P}_t^* = [P_1^*(i_{t,1}^*, \mathbf{z}_t), \dots, P_k^*(i_{t,k}^*, \mathbf{z}_t)]$ .

**Calculate Gating Weights:** We compute the final gating weights by applying the softmax function across the vector of maximum posterior scores  $\mathbf{P}_t^*$  corresponding to the  $k$  selected experts:

$$\mathcal{G}_t = \text{softmax}(\mathbf{P}_t^*) \quad (19)$$

**Final Output Calculation:** The output of the IDA-MoE layer for token  $\mathbf{u}_t$  is the weighted sum of the outputs from the  $k$  selected expert networks, using the calculated gating weights:

$$\text{MoE}(\mathbf{u}_t) = \sum_{j=1}^k \mathcal{G}_{t,j} \cdot \text{FFN}_{i_{t,j}^*}(\mathbf{u}_t) \quad (20)$$

These gating weights are derived directly from the input distribution via the GMM posteriors, ensuring that routing remains decoupled from the final task optimization, breaking the problematic feedback loop described in Section 3.2.

### 4.3 Component Reactivation Strategy

GMM training via minimizing the NLL loss (Eq.(15)) can suffer from uneven convergence rates across mixture components. Consider the gradient with respect to the mean  $\boldsymbol{\mu}_k$  of a specific component  $k = (j, i, m)$ :

$$\frac{\partial \mathcal{L}_{\text{GMM}}}{\partial \boldsymbol{\mu}_k} = - \sum_{t=1}^T \underbrace{\left( \frac{\pi_k \mathcal{N}(\mathbf{z}_t | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p(\mathbf{z}_t)} \right)}_{\mathbf{p}_{tk} \text{: Posterior}} \boldsymbol{\Sigma}_k^{-1} (\mathbf{z}_t - \boldsymbol{\mu}_k) \quad (21)$$

As shown in Eq.(21), the gradient for component  $k$  is weighted by its posterior probability given the data  $\mathbf{z}_t$ . Components initialized in sparse regions of the feature space may consistently yield very low posterior probabilities ( $\mathbf{p}_{tk} \approx 0$ ), leading to extremely slow convergence for those parameters, hindering the GMM's ability to fully capture the complete input distribution efficiently.

To accelerate the convergence of these slowly updating components, we introduce a targeted reactivation strategy. We first identify potentially inactive components signaled by their low mixing coefficient  $\pi_k$ , which often correlate with low average posterior probabilities, using a stochastic check:

$$\text{is\_slow}(k) \sim \text{Bernoulli}(\text{ReLU}(1 - N \times M \cdot \pi_k)) \quad (22)$$

For the subset  $S_j$  of components flagged as slow within each GMM set  $j$ , we apply a targeted **reactivation loss term**:

$$\mathcal{L}_{\text{react},j} = - \sum_{t=1}^T \log \left( \sum_{k \in S_j} \pi_k \mathcal{N}(\text{sg}(\mathbf{z}_t) | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right) \quad (23)$$

By calculating gradients based on Eq.(23), we mathematically **renormalize the posterior probabilities within the subset  $S_j$** . The new objective  $\mathcal{L}_{\text{react}}$  provides substantial gradient updates to slow-updating components, pulling them towards relevant data points more quickly than  $\mathcal{L}_{\text{GMM}}$  alone could, thus accelerating their convergence. Unlike conventional auxiliary losses that impose explicit balancing penalties and potentially disrupt expert specialization, this reactivation technique merely accelerates the natural convergence of the GMM towards a better fit of the input distribution. This approach preserves the specialized nature of expert routing while ensuring all components actively contribute to the model.

**Training Objectives.** The entire IDA-MoE model, including the backbone, autoencoder, and GMM parameters, is trained end-to-end by minimizing the composite loss function  $\mathcal{L}_{\text{total}}$ :

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \alpha \cdot \mathcal{L}_{\text{AE}} + \beta \cdot \sum_{j=1}^k (\mathcal{L}_{\text{GMM},j} + \mathcal{L}_{\text{react},j}) \quad (24)$$

where  $\mathcal{L}_{\text{CE}}$  is the cross-entropy loss for language modeling. The hyperparameters  $\alpha$  and  $\beta$  control the relative weight of the representation learning and distribution modeling objectives.

## 5 Experiments

### 5.1 Experimental Setup

**Model & Training Setup.** Our IDA-MoE model builds upon LLaVA architecture [37, 38]. The MoE layers use top-2 routing, a latent dimensionality of 32, and 16 GMM components per expert with diagonal covariance for efficiency. We set both  $\alpha$  and  $\beta$  to 0.01. Training follows the three-stage protocol detailed in MoE-LLaVA [30]. We bootstrap the dense model and GMMs in Stages 1 & 2 using LLaVA-1.5-558k [36], SViT [66], LVIS [52], LRV [35], and MIMIC-IT [23]. In Stage 3, we finetune IDA-MoE follow the same data pipeline as LLaVA-mix-665k [36]. Further configuration and training specifics are detailed in the Supplementary Material Section 1 & 2.

**Evaluation & Baselines.** We assess the performance across a comprehensive benchmark suite including MME [29], MMB [40], VizWiz [15], GQA [18], TextVQA [51], MM-Vet [64], ScienceQA [41], and POPE [26]. Load balance is quantified using the mean Coefficient of Variation ( $\text{CV}_{\text{mean}}$ ) of expert utilization (lower indicates better balance). For fair comparison with state-of-the-art VLMs, we selected models based on comparable parameter scales and similar training data size and methodologies. To further isolate the impact of our routing strategy, we also compare against different routing methods using identical backbone model and training data. Results for the dense baseline and MoE-LLaVA [30] are reported from [30], while other MoE implementations (xMoE [7], st-MoE [70], DeepSeek-MoE [8], AuxFree-MoE [53]) are reproduced by us for fair comparison. Metric calculation details and reproduction specifics are provided in the Supplementary Material Section 3.

**Table 1: Comparison with state-of-the-art MLLMs on the commonly-used multimodal benchmarks for MLLMs. Vis. refers to the vision encoder and image resolution, Act. refers activated parameters during inference, The best result for model sizes around 2B is shown in bold, and the second-best result is underlined.**

Method	LLM	Vis.	Act.	GQA	PoPE	VisWiz	SQA <sup>I</sup>	TextVQA	MME	MMB	MMVET
VILA-7B[31]	LLaMA-7B	CLIP-336	7B	62.3	-	57.8	68.2	64.4	1533.0	68.9	-
InstructBLIP[38]	Vicuna-13B	Vit-224	13B	49.5	-	33.4	63.1	50.7	1212.8	-	-
Qwen-VL-Chat[1]	Qwen-7B	Vit-448	7B	57.5	-	38.9	68.2	61.5	1487.5	60.6	-
LLaVA-1.5-7B[36]	Vicuna-1.5-7B	CLIP-336	7B	62.0	86.7	50.0	68.4	58.2	1476.9	61.5	30.2
Bunny-3B[17]	Phi-2-2.7B	SigLIP-384	3.1B	62.5	86.8	43.8	70.9	56.7	1488.8	68.6	-
LLaVA-Phi[69]	Phi-2-2.7B	CLIP-336	2.8B	-	85.0	35.9	68.4	48.6	1335.1	59.8	28.9
VILA-3B[31]	LLaMA-2.7B	SigLIP-384	2.7B	61.5	-	53.5	69.0	60.4	-	63.4	-
Mini-Gemini[28]	Gemma-2B	SigLIP-384	2.7B	60.7	-	41.5	63.1	56.2	1341	59.8	31.1
MoE-LLaVA[30]	Phi-2-2.7B	CLIP-336	3.6B	61.4	86.3	43.9	68.5	51.4	1423.0	65.2	34.3
Imp-2B[46]	Qwen-1.5-1.8B	SigLIP-384	1.9B	61.9	86.7	39.6	66.1	54.5	1304.8	63.8	-
Bunny-2B[17]	Qwen-1.5-1.8B	SigLIP-384	1.9B	59.6	85.8	34.2	64.6	53.2	1300.8	59.1	-
MoE-LLaVA[30]	Qwen-1.8B	CLIP-336	2.2B	61.5	<b>87.0</b>	32.6	63.1	48.0	1291.6	59.7	25.3
MoE-LLaVA[30]	Stablelm-1.6B	CLIP-336	2.0B	60.2	85.7	36.2	62.6	50.1	1318.1	60.2	26.9
IDA-MoE(ours)	Stablelm-1.6B	CLIP-336	2.0B	58.2	<u>86.8</u>	<b>43.1</b>	65.1	51.9	<u>1372.0</u>	63.2	<u>29.5</u>
IDA-MoE(ours)	Qwen2-1.5B	SigLIP-384	2.0B	<b>62.1</b>	86.8	38.6	<b>71.1</b>	<b>58.7</b>	<b>1402.8</b>	<b>66.7</b>	<b>33.4</b>

**Table 2: Performance comparison of MoE approaches. All methods use Stablelm-1.6B and CLIP-336 as base models.**

Method	MMVet	MME-per	PoPE	GQA	SQA <sup>I</sup>	TextVQA	VizWiz	MMB	CV <sub>mean</sub>
Dense	26.6	1337.3	85.6	60.6	62.7	49.9	38.9	58.2	-
MoE LLaVA[30]	26.9	1318.2	85.7	60.2	62.6	50.1	36.2	60.2	0.128
xMoE[7]	28.5	1353.8	85.6	60.9	62.7	50.5	37.3	60.1	0.117
st-MoE[70]	29.3	1364.5	85.4	60.9	61.3	50.3	38.2	59.5	0.105
DeepSeek-MoE[8]	28.7	1354.5	85.6	60.7	61.2	49.2	38.4	60.4	0.530
AuxFree-MoE[53]	29.4	1329.9	84.7	<b>61.1</b>	63.1	50.3	41.7	60.3	0.431
IDA-MoE(ours)	<b>29.5</b>	<b>1372.0</b>	<b>86.8</b>	58.3	<b>65.1</b>	<b>51.9</b>	<b>43.1</b>	<b>63.2</b>	0.143

## 5.2 Main Results

*Comparison with State-of-the-Art Models.* Table 1 presents a comprehensive comparison of IDA-MoE with recent state-of-the-art VLMs across multiple benchmarks. Despite having significantly fewer activated parameters, our 2B parameter models achieve competitive performance with much larger dense models. Our Qwen-2-1.5B variant performs well against 7B models like LLaVA-1.5-7B and Qwen-VL-Chat, while outperforming several 3B models such as LLaVA-Phi, Mini-Gemini and MoE-LLaVA-Phi on multiple benchmarks despite using only 2B activated parameters. Within the 2B parameter category, IDA-MoE with Qwen2-1.5B clearly demonstrates state-of-the-art performance, ranking first on six benchmarks and second on PoPE. While our StableLM-1.6B variant also shows strong results, particularly on VisWiz benchmark.

*Comparison Across MoE Implementations.* As shown in Table 2, we compare different MoE methods on the same LLaVA backbone to isolate the impact of routing and balancing mechanisms. Methods that enforce strict auxiliary losses, such as MoE-LLaVA, xMoE, and st-MoE, consistently produce low CV<sub>mean</sub> values, indicating excellent load balance. However, MoE-LLaVA, which adopts standard auxiliary loss, shows only marginal performance gains over the dense baseline. st-MoE introduces router-z loss to stabilize training, and xMoE employs dimensionality reduction combined with

cosine-similarity-based routing to address representation collapse. Although these enhancements allow st-MoE and xMoE to yield better results than the vanilla MoE setup, their fundamental reliance on the auxiliary balancing objective still prevents them from fully exploiting the potential capacity benefits inherent in the MoE architecture. In contrast, DeepSeek-MoE uses a significantly smaller load balancing coefficient (0.001 vs. the typical 0.01), while AuxFree-MoE eliminates auxiliary losses entirely, relying instead on prior distribution correction for load balancing. These methods relax balancing constraints to yield better task performance but suffer from substantial load imbalance (high CV<sub>mean</sub>). IDA-MoE stands out by achieving superior performance across most benchmarks while maintaining effective load balance (CV<sub>mean</sub>: 0.1437) without relying on auxiliary losses. This demonstrates the effectiveness of our distribution-driven, decoupled routing strategy in mitigating the specialization-balance dilemma.

## 5.3 Ablation Study

*Impact of routing dimensionality.* Table 3 illustrates impact of the routing space dimensionality on model performance and load balance. First, we observe that routing requires sufficiently high dimensionality to maintain discriminative power. While lower dimensions (4-8) produce slightly more balanced expert utilization,

**Table 3: Ablation study on dimensionality reduction.**

Centers	Dim	MME	TextVQA	VizWiz	SQA <sup>I</sup>	CV <sub>mean</sub>
16	4	1340.7	49.95	36.77	61.73	0.1391
16	8	1363.4	50.53	36.84	62.22	0.1430
16	16	1365.4	50.40	38.40	62.82	0.1434
16	32	<b>1372</b>	<b>51.86</b>	<b>43.11</b>	<b>65.05</b>	0.1437
16	64	1364.1	50.29	40.61	63.61	0.1492

they significantly compromise performance across all benchmarks. This suggests that overly simplified representations lack the capacity to effectively distinguish between different input patterns, which make the token allocation more even but also leading to suboptimal expert assignment. Second, we find that performance gains saturate and eventually decline as dimensionality increases beyond an optimal point. This aligns with the well-known curse of dimensionality in mixture models, where high-dimensional spaces become increasingly sparse, making density estimation less reliable for routing decisions. This results identify 32 dimensions as the sweet spot that maximizes model performance while maintaining effective load distribution across experts and minimum overhead.

**Table 4: Ablation study on the number of GMM centers.**

Centers	Dim	MME	TextVQA	VizWiz	SQA <sup>I</sup>	CV <sub>mean</sub>
1	32	1329.83	49.69	38.56	63.01	0.3218
4	32	1344.62	49.83	39.91	63.41	0.1974
16	32	<b>1372</b>	<b>51.86</b>	<b>43.11</b>	<b>65.05</b>	0.1437
32	32	1340.31	49.43	36.18	63.51	0.1364

*Impact of GMM components per expert.* Table 4 presents the impact of varying the number of GMM centers for the IDA-MoE model. The 16-center configuration achieves substantial improvements over the single-center baseline: +42.17 points on MME, +2.17 points on TextVQA, +4.55 points on VizWiz, and +2.04 points on SQA. These consistent gains across all benchmarks validate the effectiveness of multi-center approach for modeling token distributions. However, further increasing the number of centers to 32 leads to performance degradation across all tasks, suggesting a potential overfitting effect where additional components may capture noise rather than meaningful token patterns.

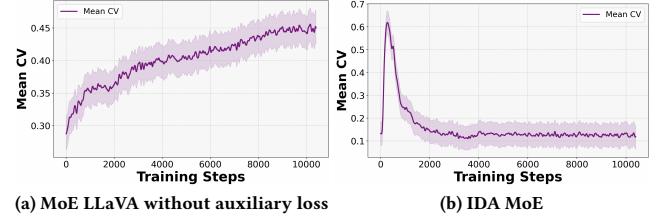
The CV<sub>mean</sub> metric shows a consistent decrease as the number of centers increases, dropping from 0.3218 (1 center) to 0.1364 (32 centers). This indicates that a finer-grained partitioning of the input space leads to more uniform expert utilization. Although the 32-center configuration yields the most balanced expert workload (lowest CV<sub>mean</sub>), the 16-center configuration achieves the optimal balance between maximizing task performance and significantly improving expert utilization compared to configurations with fewer centers.

**Table 5: Ablation study on the effect of Component Reactivation Strategy.**

Re.act.	Centers	MME	TextVQA	VizWiz	SQA <sup>I</sup>	CV <sub>mean</sub>
✓	16	<b>1372</b>	<b>51.86</b>	<b>43.11</b>	65.05	0.1437
✗	16	1360.88	51.51	41.36	<b>65.54</b>	0.1544

*Impact of component reactivation strategy.* Table 5 analyzes the impact of the Reactivation Strategy for the IDA-MoE model (StableLM 1.6B backbone, 16 centers). The results show that incorporating Reactivation yields performance improvements on several benchmarks, increasing the MME score by +11.12 points, TextVQA accuracy by +0.35 points, and VizWiz accuracy by +1.75 points. Although a slight decrease of 0.49 points is observed on SQA<sup>I</sup>, the strategy demonstrates a generally positive impact on model performance across the evaluated datasets. Furthermore, as the Reactivation Strategy is designed to accelerate the convergence of input distribution modeling, we can see a further improved CV<sub>mean</sub> value from 0.1544 to 0.1437. These findings suggest the Reactivation Strategy's overall positive contribution to both task performance and expert load balancing for MoE routing.

## 5.4 Quantitative Analysis

**Figure 5: CV<sub>mean</sub> change over training.**

*Decoupling Routing From Task-Specific Loss Leads to Natural Load Balance.* As analyzed in Section 3.2, conventional routing mechanisms exhibit a persistent tendency toward imbalanced expert utilization. As illustrated in Fig. 5a, traditional MoE Architecture without auxiliary loss exhibits steadily increasing CV<sub>mean</sub> values, indicating progressively worsening load imbalance as training goes. In contrast, by decoupling the training of token allocation from task performance, Fig. 5b demonstrates naturally declining CV<sub>mean</sub> values throughout the training without introducing explicit auxiliary balancing losses. This emergent load-balancing behavior arises from the nature of convergence Gaussian Mixture Modeling of the input space. IDA-MoE establishes a more principled foundation for expert specialization while maintaining balanced utilization, all without the need for heuristic regularization terms that can potentially compromise task performance.

## 6 Conclusion

In this paper, we introduce Input Domain Aware Mixture of Experts (IDA-MoE), a novel approach that fundamentally rethinks expert routing in vision language models. By decoupling the router training from task-specific loss functions and employing a principled probabilistic model for token allocation, IDA-MoE naturally strikes a balance between load distribution and model performance without relying on auxiliary losses or heuristic constraints. Empirical results demonstrate that IDA-MoE consistently outperforms previous routing and load balancing mechanisms across a diverse set of benchmarks. This performance advantage stems from more decisive routing decisions and better-specialized experts, while maintaining excellent load balance.

## Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant No. 62276245).

## References

- [1] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609* (2023).
- [2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923* (2025).
- [3] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhajit Som, Songhao Piao, and Furu Wei. 2022. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Advances in Neural Information Processing Systems* 35 (2022), 32897–32912.
- [4] Haoyu Cao, Changcun Bao, Chaohu Liu, Huang Chen, Kun Yin, Hao Liu, Yinsong Liu, Deqiang Jiang, and Xing Sun. 2023. Attention Where It Matters: Rethinking Visual Document Understanding with Selective Region Concentration. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1–6, 2023*. IEEE, 19460–19470. doi:10.1109/ICCV51070.2023.01788
- [5] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. 2024. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271* (2024).
- [6] Zhe Chen, Jianne Wu, Wenhui Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 24185–24198.
- [7] Zewen Chi, Li Dong, Shaohan Huang, Damai Dai, Shuming Ma, Barun Patra, Saksham Singhal, Payal Bajaj, Xia Song, Xian-Ling Mao, et al. 2022. On the representation collapse of sparse mixture of experts. *Advances in Neural Information Processing Systems* 35 (2022), 34600–34613.
- [8] Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y Wu, et al. 2024. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066* (2024).
- [9] Damai Dai, Li Dong, Shuming Ma, Bo Zheng, Zhifang Sui, Baobao Chang, and Fur Wei. 2022. Stablemoe: Stable routing strategy for mixture of experts. *arXiv preprint arXiv:2204.08396* (2022).
- [10] William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research* 23, 120 (2022), 1–39.
- [11] Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Yuhang Dai, Meng Zhao, Yi-Fan Zhang, Shaoqi Dong, Yangze Li, Xiong Wang, et al. 2024. Vita: Towards open-source interactive omni multimodal llm. *arXiv preprint arXiv:2408.05211* (2024).
- [12] Chaoyou Fu, Haojia Lin, Xiong Wang, Yi-Fan Zhang, Yunhang Shen, Xiaoyu Liu, Haoyu Cao, Zuwei Long, Heting Gao, Ke Li, et al. 2025. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *arXiv preprint arXiv:2501.01957* (2025).
- [13] Zixian Gao, Disen Hu, Xun Jiang, Huimin Lu, Heng Tao Shen, and Xing Xu. 2024. Enhanced Experts with Uncertainty-Aware Routing for Multimodal Sentiment Analysis. In *ACM Multimedia 2024*. <https://openreview.net/forum?id=78TMql1c04>
- [14] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).
- [15] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3608–3617.
- [16] Jiaao He, Jidong Zhai, Tiago Antunes, Haojie Wang, Fuwen Luo, Shangfeng Shi, and Qin Li. 2022. FasterMoE: modeling and optimizing training of large-scale dynamic pre-trained models. In *Proceedings of the 27th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming* (Seoul, Republic of Korea) (PPoPP '22). Association for Computing Machinery, New York, NY, USA, 120–134. doi:10.1145/3503221.3508418
- [17] Muyang He, Yexin Liu, Boya Wu, Jianhao Yuan, Yueze Wang, Tiejun Huang, and Bo Zhao. 2024. Efficient multimodal learning from data-centric perspective. *arXiv preprint arXiv:2402.11530* (2024).
- [18] Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 6700–6709.
- [19] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088* (2024).
- [20] Yechan Kim, Hwijoong Lim, and Dongsu Han. 2024. Scaling Beyond the GPU Memory Limit for Large Mixture-of-Experts Model Training. In *Proceedings of the 41st International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 235)*. Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (Eds.). PMLR, 24342–24353. <https://proceedings.mlr.press/v235/kim24w.html>
- [21] Aran Komatsuzaki, Joaquin Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. 2022. Sparse upcycling: Training mixture-of-experts from dense checkpoints. *arXiv preprint arXiv:2212.05055* (2022).
- [22] Dmitry Lepikhin, Hyoukjoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2020. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668* (2020).
- [23] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Fanyi Pu, Jingkang Yang, Chunyuan Li, and Ziwei Liu. 2023. Mimic-it: Multi-modal in-context instruction tuning. *arXiv preprint arXiv:2306.05425* (2023).
- [24] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*. PMLR, 19730–19742.
- [25] Jiachen Li, Xinyao Wang, Sijie Zhu, Chia-Wen Kuo, Lu Xu, Fan Chen, Jitesh Jain, Humphrey Shi, and Longyin Wen. 2024. Cumo: Scaling multimodal llm with co-upcycled mixture-of-experts. *Advances in Neural Information Processing Systems* 37 (2024), 131224–131246.
- [26] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355* (2023).
- [27] Yinxin Li, Shenyuan Jiang, Baotian Hu, Longyue Wang, Wanqi Zhong, Wenhan Luo, Lin Ma, and Min Zhang. 2025. Uni-moe: Scaling unified multimodal llms with mixture of experts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025).
- [28] Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. 2024. Mini-gemini: Mining the potential of multi-modality vision language models. *arXiv preprint arXiv:2403.18814* (2024).
- [29] Zijing Liang, Yanjie Xu, Yifan Hong, Penghui Shang, Qi Wang, Qiang Fu, and Ke Liu. 2024. A Survey of Multimodel Large Language Models. In *Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering*. 405–409.
- [30] Bin Lin, Zhenyu Tang, Yang Ye, Jiaxi Cui, Bin Zhu, Peng Jin, Jinfu Huang, Junwu Zhang, Yatian Pang, Munan Ning, et al. 2024. Moe-llava: Mixture of experts for large vision-language models. *arXiv preprint arXiv:2401.15947* (2024).
- [31] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. 2024. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 26689–26699.
- [32] Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Daya Guo, et al. 2024. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434* (2024).
- [33] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437* (2024).
- [34] Chaohua Liu, Kun Yin, Haoyu Cao, Xinghua Jiang, Xin Li, Yinsong Liu, Deqiang Jiang, Xing Sun, and Linli Xu. 2024. HRVDA: High-Resolution Visual Document Assistant. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16–22, 2024*. IEEE, 15534–15545. doi:10.1109/CVPR52733.2024.01471
- [35] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. 2023. Aligning Large Multi-Modal Model with Robust Instruction Tuning. *arXiv preprint arXiv:2306.14565* (2023).
- [36] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023. Improved Baselines with Visual Instruction Tuning. *arXiv preprint arXiv:2310.03744* (2023).
- [37] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 26296–26306.
- [38] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems* 36 (2023), 34892–34916.
- [39] Liyuan Liu, Young Jin Kim, Shuohang Wang, Chen Liang, Yelong Shen, Hao Cheng, Xiaodong Liu, Masahiro Tanaka, Xiaoxia Wu, Wenxiang Hu, et al. 2024. Grin: Gradient-informed moe. *arXiv preprint arXiv:2409.12136* (2024).
- [40] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2024. Mmbench: Is your multi-modal model an all-around player?. In *European conference on computer vision*. Springer, 216–233.

[41] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multi-modal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems* 35 (2022), 2507–2521.

[42] Xiaoman Nie, Xupeng Miao, Zilong Wang, Zichao Yang, Jilong Xue, Lingxiao Ma, Gang Cao, and Bin Cui. 2023. Flexmoe: Scaling large-scale sparse pre-trained model training via dynamic device placement. *Proceedings of the ACM on Management of Data* 1, 1 (2023), 1–19.

[43] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.

[44] Xiaozhe Ren, Pingyi Zhou, Xinfan Meng, Xinxing Huang, Yadao Wang, Weichao Wang, Pengfei Li, Xiaoda Zhang, Alexander Podolskiy, Grigory Arshinov, et al. 2023. Pangu- $\{\Sigma\}$ : Towards trillion parameter language model with sparse heterogeneous computing. *arXiv preprint arXiv:2303.10845* (2023).

[45] Stephen Roller, Sainbayar Sukhbaatar, Jason Weston, et al. 2021. Hash layers for large sparse models. *Advances in Neural Information Processing Systems* 34 (2021), 17555–17566.

[46] Zhenwei Shao, Zhou Yu, Jun Yu, Xuecheng Ouyang, Lihao Zheng, Zhenbiao Gai, Mingyang Wang, and Jiajun Ding. 2024. Imp: Highly capable large multimodal models for mobile devices. *arXiv preprint arXiv:2405.12107* (2024).

[47] Noam Shazeer, \*Azalia Mirhoseini, \*Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=B1ckMDqlg>

[48] Yunhang Shen, Chaoyou Fu, Shaqiq Dong, Xiong Wang, Yi-Fan Zhang, Peixian Chen, Mengdan Zhang, Haoyu Cao, Ke Li, Xiawu Zheng, et al. 2025. Long-VITA: Scaling Large Multi-modal Models to 1 Million Tokens with Leading Short-Context Accuracy. *arXiv preprint arXiv:2502.05177* (2025).

[49] Yikang Shen, Zheyu Zhang, Tianyou Cao, Shawn Tan, Zhenfang Chen, and Chuang Gan. 2023. ModuleFormer: Modularity Emerges from Mixture-of-Experts. *arXiv preprint arXiv:2306.04640* (2023).

[50] Fangxun Shu, Yue Liao, Le Zhuo, Chenning Xu, Lei Zhang, Guanghao Zhang, Haonan Shi, Long Chen, Tao Zhong, Wanggui He, et al. 2024. Llava-mod: Making llava tiny via moe knowledge distillation. *arXiv preprint arXiv:2408.15881* (2024).

[51] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8317–8326.

[52] Junke Wang, Lingchen Meng, Zejia Weng, Bo He, Zuxuan Wu, and Yu-Gang Jiang. 2023. To see is to believe: Prompting GPT-4V for better visual instruction tuning. *arXiv preprint arXiv:2311.07574* (2023).

[53] Lean Wang, Huazuo Gao, Chenggang Zhao, Xu Sun, and Damai Dai. 2024. Auxiliary-loss-free load balancing strategy for mixture-of-experts. *arXiv preprint arXiv:2408.15664* (2024).

[54] Wei Wang, Zhiqian Lai, Shengwei Li, Weijie Liu, Keshi Ge, Ao Shen, and Dongsheng Li. 2024. Pro-Prophet: Systematic Load Balancing Method for Efficient Parallel Training of Large-scale MoE Models. *arXiv preprint arXiv:2411.10003* (2024).

[55] Tianwen Wei, Bo Zhu, Liang Zhao, Cheng Cheng, Biye Li, Weiwei Lü, Peng Cheng, Jianhao Zhang, Xiaoyu Zhang, Liang Zeng, et al. 2024. Skywork-moe: A deep dive into training techniques for mixture-of-experts language models. *arXiv preprint arXiv:2406.06563* (2024).

[56] Chenwei Wu, Zitao Shuai, Zhengxu Tang, Luning Wang, and Liyue Shen. [n. d.]. Dynamic Modeling of Patients, Modalities and Tasks via Multi-modal Multi-task Mixture of Experts. In *The Thirteenth International Conference on Learning Representations*.

[57] Haoze Wu, Zihan Qiu, Zili Wang, Hang Zhao, and Jie Fu. 2024. GW-MoE: Resolving Uncertainty in MoE Router with Global Workspace Theory. *arXiv preprint arXiv:2406.12375* (2024).

[58] Jialin Wu, Xia Hu, Yaqing Wang, Bo Pang, and Radu Soricut. 2024. Omni-smola: Boosting generalist multimodal models with soft mixture of low-rank experts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14205–14215.

[59] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, et al. 2024. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302* (2024).

[60] Wenxin Xu, Hexin Jiang, and xuefeng liang. 2024. Leveraging Knowledge of Modality Experts for Incomplete Multimodal Learning. In *ACM Multimedia 2024*. <https://openreview.net/forum?id=Gt3a8A1wLg>

[61] Fuzhao Xue, Zian Zheng, Yao Fu, Jinjin Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. 2024. Openmoe: An early effort on open mixture-of-experts language models. *arXiv preprint arXiv:2402.01739* (2024).

[62] Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Guoyin Wang, Heng Li, Jiangcheng Zhu, Jianqun Chen, et al. 2024. Yi: Open foundation models by 01. *ai*. *arXiv preprint arXiv:2403.04652* (2024).

[63] Haofei Yu, Zhengyang Qi, Lawrence Jiang, Ruslan Salakhutdinov, Louis-Philippe Morency, and Paul Pu Liang. 2023. MMoE: Enhancing Multimodal Models with Mixtures of Multimodal Interaction Experts. *arXiv preprint arXiv:2311.09580* (2023).

[64] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2023. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490* (2023).

[65] Haotian Zhang, Mingfei Gao, Zhe Gan, Philipp Dufter, Nina Wenzel, Forrest Huang, Dhruti Shah, Xianzhi Du, Bowen Zhang, Yanghao Li, et al. 2024. Mml. 5: Methods, analysis & insights from multimodal llm fine-tuning. *arXiv preprint arXiv:2409.20566* (2024).

[66] Bo Zhao, Boya Wu, and Tiejun Huang. 2023. Svit: Scaling up visual instruction tuning. *arXiv preprint arXiv:2307.04087* (2023).

[67] Yanqi Zhou, Nan Du, Yanping Huang, Daiyi Peng, Chang Lan, Da Huang, Siyam Shakheri, David So, Andrew M Dai, Yifeng Lu, et al. 2023. Brainformers: Trading simplicity for efficiency. In *International Conference on Machine Learning*. PMLR, 42531–42542.

[68] Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. 2022. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems* 35 (2022), 7103–7114.

[69] Yichen Zhu, Minjie Zhu, Ning Liu, Zhiyuan Xu, and Yixin Peng. 2024. Llava-phi: Efficient multi-modal assistant with small language model. In *Proceedings of the 1st International Workshop on Efficient Multimedia Computing under Limited*. 18–22.

[70] Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. 2022. St-moe: Designing stable and transferable sparse expert models. *arXiv preprint arXiv:2202.08906* (2022).

[71] Simiao Tian, Xiaodong Liu, Jian Jiao, Young Jin Kim, Hany Hassan, Ruofei Zhang, Tuo Zhao, and Jianfeng Gao. 2021. Taming sparsely activated transformer with stochastic experts. *arXiv preprint arXiv:2110.04260* (2021).